

and Rosa Maria Viccari). Map making as a support for cooperation (Fumihiko Chimura and Mario Tokoro). Distributed flow shop scheduling problem—Global versus local optimization (Thouraya Daouas, Khaled Ghedira, and Jean Pierre Muller). The consensus scheduling model: Negotiation among peers (William B. Day). Actions with typical effects: Epistemic characterization of scenarios (Barbara Dunin-Keplicz, and Anna Radzikowska). Unifying control in a layered agent architecture (Klaus Fischer, Jörg P. Muller, and Markus Pischel). Deriving goals and intentions (Graça Gaspar and Helder Coelho). Agents as reasoners, observers, or arbitrary believers (Enrico Giunchiglia, Fausto Giunchiglia, and Luciano Serafini). A quantitative analysis of the contract net protocol (Cheng Gu and Toru Ishida). Evolving cooperation strategies (Thomas Haynes, Roger Wainwright, and Sandip Sen). Distributed symbolic-subsymbolic agent architecture for configuring power network faults (R. Khosla and T. Dillon). An agent algebra for the formal description and verification of multi agent systems (Alexander Kick). Coalition-based cooperation between intelligent agents for a contextual recognition of interdatabase dependencies (Matthias Klusch). Self-organization based on coordinated actions of autonomous agents (Satoshi Kurihara and Michio Okada). AgenTalk: Coordination protocol description for multiagent systems (Kazuhiro Kuwabara, Toru Ishida, and Nobuyasu Osato). Multiagent coordination in distributed interactive battlefield simulations (John E. Laird, Randolph M. Jones, and Paul E. Nielsen). A microeconomic approach to intelligent resource sharing in multiagent systems (Jaeho Lee and Edmund H. Durfee). A view on the architecture and design of highly autonomous and situated agents (J. Masthoff and R. Van Hoe). Organic programming for multi-agents (Hideyuki Nakashima, Itsuki Noda, and Ichiro Ohsawa). A multi-agent approach to the design of coordination mechanisms (Carla Simone, Monica Divitini, Kjeld Schmidt, and Peter Cartersen). Some principles in designing cooperative architectures of autonomous agents (Lorenzo Sommaruga and Nadia Catenazzi). Learning coordination plans in distributed problem solving environments (Toshiharu Sugawara and Victor Lesser). An agent model in a multiagent system architecture for automating distributed systems (Walt Truszkowski, Jidé Odubiyi, and Ed Ruberton). Beyond cooperation-ism and competition-ism (exploring social phenomena with a generalized tit-for-tat model) (Elpida S. Tzafestas). VIENA: A multiagent interface to a virtual environment (I. Wachsmuth, B. Lenzmann, and Y. Cao). Automated multiagent preference aggregation using fuzzy quantifiers (R.R. Yager). Asynchronous weak-commitment search for solving large-scale distributed constraint satisfaction problems (M. Yokoo). The consensus of uncertainties in distributed expert systems (M. Zhang and C. Zhang). Index.

*Computation and Intelligence: Collected Readings.* Edited by George F. Luger. AAAI Press, Menlo Park, CA/ MIT Press, Cambridge, MA. (1995). 735 pages. \$35.00.

Contents:

Acknowledgments. Introduction (George Luger). I. Foundations. 1. The prehistory of android epistemology (C. Glymour, K. Ford, and P. Hayes). 2. Computing machinery and intelligence (A. Turing). 3. Steps towards artificial intelligence (M. Minsky). 4. Computer science as empirical enquiry: Symbols and search (A. Newell and H.A. Simon). II. Knowledge representation. 5. The knowledge level. 6. A framework for representing knowledge (M. Minsky). 7. Retrieval time from semantic memory (A.M. Collins and M.R. Quillian). 8. A procedural model of language understanding (T. Winograd). 9. The structure of episodes in memory. (R. Schank). 10. In defense of logic (P. Hayes). 11. Escaping brittleness: The possibility of general purpose learning algorithms applied to parallel rule-based systems (J. Holland). 12. The appeal of parallel distributed processing (J.L. McClelland, D.E. Rumelhart, and G.E. Hinton). 13. Intelligence without representation (R.A. Brooks). III. Weak method problem solving. 14. Empirical explorations with the logic theory machine (A. Newell, J.C. Shaw, and H.A. Simon). 15. Some studies in machine learning using the game of checkers (A.L. Samuel). 16. GPS, a program that simulates human thought (A. Newell and H.A. Simon). 17. STRIPS: A new approach in the application of theorem proving to problem solving (R.E. Fikes and N.J. Nilsson). 18. Blackboard systems: The blackboard model of problem solving and the evolution of blackboard architectures (H.P. Nii). IV. Reasoning in complex and dynamic environments. 19. Programs with common sense. Dendral and Meta-Dendral: Their applications dimension (B.G. Buchanan and E.A. Feigenbaum). 21. Consultation systems for physicians (E. Shortliffe). 22. A truth maintenance system (J. Doyle). 23. Circumscription—A form of non-monotonic reasoning (J. McCarthy). 24. The second naive physics manifesto (P. Hayes). 25. Paradigms for machine learning (J.G. Carbonell). 26. Improving human decision making through case based decision aiding (J. Kolodner). 27. Diagnosis based on description of structure and function (R. Davis, H. Shrobe, W. Hamscher, K. Wieckert, M. Shirley, and S. Polit). 28. Pengi: An implementation of a theory of activity (P. Agre and D. Chapman). V. Foundations revisited. 29. Logical versus analogical or symbolic versus connectionist or neat versus scruffy (M. Minsky). 30. Machine as mind (H.A. Simon). Bibliography. Index.

*The Capability Maturity Model: Guidelines for Improving the Software Process.* Edited by Mark Paulk, Charles V. Weber, Bill Curtis, and Mary Beth Chrissis. Addison-Wesley, Reading, MA. (1995). 441 pages. \$49.44.

Contents:

I. The capability maturity model for software: Background, concepts, structure, and usage. 1. Introducing software process maturity. 2. The software process maturity framework. 3. The structure of the capability maturity model. 4. Interpreting the CMM. 5. Using the CMM. 6. A high-maturity example: Space shuttle onboard software. II. The key practices of the capability maturity model for software. 7. The key areas for Level 2: Repeatable. 8. The key process areas for Level 3: Defined. 9. The key process areas for Level 4: Managed. 10. The key process areas for Level 5: Optimizing. Appendixes. A. References. B. Acronyms. C. Glossary. D. Abridged version of the key practices. E. Mapping the key practices to goals. F. Comparing ISO 9001 and the CMM. G. An overview of ISO's SPICE project. H. Change history of the CMM. I. Change Request Form. Index.